

Application No. 09/541,631 (Balkany) GAU 2172

Page 4 of 12

CLAIMS: *The following is a listing of all claims in the application with their status and the text for all active claims.*

- A5
1. (CURRENTLY AMENDED) A method for storing a plurality of parallel data element sequences comprising ~~the steps of~~:
 - (a) creating a dictionary of unique values for each of said data element sequences, whereby wherein each dictionary associates contains a numeric index with for each unique value in the corresponding sequence;
 - (b) forming an n-ary tree with leaf and interior nodes ~~where~~ wherein:
 - (1) each said leaf node corresponds to one of said dictionaries,
 - (2) each said interior node associates a numeric index with tuples of numeric indexes from other subordinate leaf or interior nodes, and
 - (3) interior nodes ~~may store~~ are capable of storing one or more sequences of unique, mutually consecutive tuples separately from the other tuples using the length of said sequence.
 2. (CURRENTLY AMENDED) The method of claim 1, ~~whereby~~ wherein each unique value of a leaf node ~~and or~~ each unique tuple of an interior node is associated with a count of the number of times that value or ~~implied~~ tuple of values occurred in the parallel data element sequences.
 3. (CURRENTLY AMENDED) The method of claim 1, further including a means for efficiently processing a subset of a tree's leaves, comprising the following steps:
 - (a) ~~defining the definition of~~ a gate field in each interior nodes node,
 - (b) ~~setting each of said gate field's values~~ the value of said gate field in each said interior node, to indicate which of ~~the corresponding said~~ interior node's branches lead to leaf nodes in said subset,
 - (c) following paths that lead to said leaves leaf nodes, and
 - (d) ~~processing the leaves~~ said leaf nodes encountered.
 4. (CURRENTLY AMENDED) The method of claim 1, further including selectively disabling ~~separate~~ storage of tuple runs using the sequence length at certain interior nodes.
 5. (CURRENTLY AMENDED) The method of claim 1 further including ~~the a~~ method for arranging said n-ary tree comprising the steps of:
 - (a) ~~defining~~ a problem space ~~consisting of~~ comprising:
 - (1) a set of states such that each state contains a set of leaves and zero or more interior nodes, each with two or more other nodes as children,
 - (2) a value function, giving a numeric ranking of the value of any state's design
 - (b) defining one or more operators that transform one state to another, and

Application No. 09/541,631 (Balkany) GAU 2172

Page 5 of 12

A5

- (c) searching the problem space, starting from an initial state and applying operators to move to other states until a state with an acceptable n-ary tree design is reached.
6. (CURRENTLY AMENDED) A method for storing a plurality of parallel data element sequences comprising the steps of:
- (a) creating a dictionary of unique values for each of said data element sequences to form a set of dictionaries, whereby wherein each dictionary associates contains a numeric index with for each unique value in the corresponding sequence
 - (b) forming an one or more n-ary tree trees with leaf and interior nodes where:
 - (1) said leaf nodes are distinct from said dictionaries and each said leaf node represents is capable of representing a subset values from one of said dictionaries using numeric indexes into said dictionary, and
 - (2) each said interior node associates a numeric index with tuples of numeric indexes from other terminal or non-terminal subordinate leaf or interior nodes,wherein the one or more n-ary trees are capable of sharing said dictionaries.
7. (CURRENTLY AMENDED) The method of claim 6, ~~whereby~~ wherein each unique value of a leaf node ~~and~~ or each unique tuple of an interior node is associated with a count of the number of times that value or implied tuple of values occurred in the parallel data element sequences.
8. (CURRENTLY AMENDED) The method of claim 6, further including a means for efficiently processing a subset of a tree's leaves, comprising the following steps:
- (a) defining the definition of a gate field in each interior nodes node,
 - (b) setting each of said gate field's values the value of said gate field in each said interior node, to indicate which of the corresponding said interior node's branches lead to leaf nodes in said subset,
 - (c) following paths that lead to said leaves leaf nodes, and
 - (d) processing the leaves said leaf nodes encountered.
9. (CURRENTLY AMENDED) The method of claim 6, ~~whereby~~ further including creating an additional tree, t, is created using a subset of the same fields of the first tree, f, comprising the steps of:
- (a) finding an ancestor node in tree f, of the leaf nodes in f corresponding to said subset of fields;
 - (b) looking up the tokens of said leaf nodes corresponding to a said subset of tokens in said ancestor;
 - (c) inserting said leaf node tokens into tree, t.
10. (CURRENTLY AMENDED) The method of claim 6 further including ~~the a~~ a method for arranging said n-ary tree comprising the steps of:
- (a) defining a problem space ~~consisting of~~ comprising:

Application No. 09/541,631 (Balkany) GAU 2172

Page 6 of 12

A5

- (1) a set of states such that each state contains a set of leaves and zero or more interior nodes, each with two or more other nodes as children,
- (2) a value function, giving a numeric ranking of the value of any state's design
- (b) defining one or more operators that transform one state to another, and
- (c) searching the problem space, starting from an initial state and applying operators to move to other states until a state with an acceptable n-ary tree design is reached.

A6

- 11. (NEW) The method of claim 1, where said interior nodes are capable of storing one or more of said tuple sequences using a single tuple in combination with said sequence length.
- 12. (NEW) The method of claim 1, where one or more of said sequences is a sequence of mutually-consecutive tuples.
- 13. (NEW) The method of claim 6, where said numeric indexes are an array of counts, wherein each count is the number of times the corresponding dictionary value at the same index has occurred.
- 14. (NEW) The method of claim 6, where said numeric indexes are a bit array, wherein each bit in said array tells the presence of the corresponding dictionary value at the same index, in said parallel data element sequences.
- 15. (NEW) The method of claim 6, where said leaf node is capable of representing a subset of said dictionary's values, using a subset of numeric indexes from said dictionary.
- 16. (NEW) The method of claim 5, where said method uses an estimate of interior node size, from a function of the sizes of said interior node's child nodes.
- 17. (NEW) The method of claim 10, where said method uses an estimate of interior node size, from a function of the sizes of said interior node's child nodes.